REMARKS

Claims 2, 4, 7-9, 11, 15, and 17-55 are pending in the present application. Claims 1, 3, 5, 6, 10, 12 -14 and 16 have been canceled above in order to expedite prosecution. Claims 2, 4, 7, 15, 22, 40, 41 and 53 have been amended above to clarify the claim or to more accurately reflect the specification, and not to redirect claims to either new matter or to limit them in view of the art which is not directed to methods as claimed in the present application.

Claims 2, 4, 7-9, 11, 15, 17-39, 41-52 and 54-55 stand rejected under 35 U.S.C. §102(e) as unpatentable over U.S.P.N 5,490,097 issued to Swenson et al. ("Swenson").

Independent Claim 2 recited, "... instantiating a plurality of objects by abstract or concrete classes,..." Since the definition of an abstract class explicitly precludes the instantiation of objects and the specification provided no such disclosure, the claim has been amended to more clearly and accurately reflect the specification and the accepted use of the terms of object technology. Additional language has been added to more clearly distinguish between model entities and the real world entities that are modeled and to explicitly claim an equivalent.

Applicant believes that the clarifications afforded by the amended language make the claim patentably distinguishable over Swenson which does not describe the recited, "customizing one or more abstract classes," nor the recited, "concrete decision and data classes," nor the recited basis for, "establishing an interdependence between," one decision class and another, nor the recited, "providing a user of said method for modeling with an ability to generate additional subclasses of said abstract classes."

Claims 4 and 11 are patentable at least for the reasons discussed above in that each of these claims depends directly from Claim 2.

The Rejection of amended, and now independent, Claim 7 is respectfully

traversed on grounds that it recites, "… requiring said nodes to support participation of multiple

persons in differentiated roles in each said abstract decision situations, … ." The Examiner points

to Swenson's Fig. 1, which figure Swenson describes as,"… a diagram <u>illustrating a work process</u>

and the communications involved in <u>a typical work process</u> for processing a detected bug. [column

3, lines 40-42, underscore added]" There is no model described in Figure 1, much less the more

specific, "network whose nodes are abstract decision situations" recited by the claim, so it is

unclear how entities depicted in figure 1 support rejection of model entities.

The Examiner asserts, "… that in Swenson, the Submitter or Tester or Programmer

<u>has a role</u> because each one is responsible for a task. [Advisory action mailed 1/15/04, underscore

added]" However, the Examiner's assertion has no bearing on the issue of anticipation here,

because the Examiner is <u>pointing to a description of real world entities</u> (i.e., Submitter, Tester,

Programmer) <u>to support alleged anticipation of modeled entities</u> (i.e.., roles). Modeled entities often

have real world counterparts. The utility of models is derived, at least in part, from their mapping to

aspects of the real world. But <u>model entities are not anticipated by real world entities that they</u>

<u>model</u>, because model entities are abstractions, and abstractions are selective.

> "A model is an <u>abstract representation of reality that excludes much</u> of the world's
> infinite detail. The purpose of a model is to reduce the complexity … by eliminating
> the detail that does not influence its relevant behavior. Therefore, <u>a model reveals</u>
> <u>what its creator believes is important</u> …. [Curtis, Bill, et al., "Process Modeling,"
> Communications of the ACM, Vol. 35, No. 9, September 1992, p.76, underscore
> added]"

> "<u>When we abstract some of the essential persisting features from the specific acts</u>
> <u>comprising role behavior we speak of roles</u>. For example, we can speak of <u>the role</u>
> of the quarterback <u>on a football team</u>, in general terms of play selection <u>without</u>
> <u>specifying the particular signals he barks to his teammates or the specific plays with</u>
> <u>which they respond</u>. [Katz, Daniel and Kahn, Robert L., *The Social Psychology Of*
> *Organizations*, Wiley: New York, NY, 1966., p. 174, underscore added]]"

> "Abstraction is the <u>selective</u> examination of certain aspects of a problem. The goal
> of abstraction is to isolate those aspects that are important for some purpose and

suppress those aspects that are unimportant. Abstraction must always be for some purpose, because the purpose determines what is or is not important. <u>Many different abstractions of the same thing are possible</u>, depending on the purpose for which they are made. [Rumbaugh, James, *et al.*, *Object-Oriented Modeling and Design*, Prentice-Hall: Englewood Cliffs, NJ, 1991, p. 16, underscore added]"

The Examiner's rejection rests upon failure to distinguish between a real world entity and an abstract entity. Claim 7 does not recite that someone "has a role," but rather, that nodes of a network support participation of multiple persons in differentiated roles. Such modeling support is nowhere described in Swenson.

Furthermore, the roles which the Examiner alleges to be described in Swenson (i.e., Submitter, Tester, Programmer, Project Manager, Tester 1, Tester 2 or Test Lead) are actually described in Swenson, not as roles, but as <u>individuals</u>.

"FIG. 1 shows a diagram of a work process for processing a bug detected in some software. <u>The process involves four individual users</u>, a submitter 10, a tester 12, a programmer 14 and a project manager 16 [Column 4, line 65 - column 5, line 1, underscore added]."

In addition, a role is related to a context (e.g., in a *movie*, in an *organization*, in a *process*, in an *activity*). For example, like the quarterback role <u>on a football team</u>,

"The various members of a family interact in consistent ways in assuming <u>their role</u> <u>[in the family]</u> as father, mother , son, daughter, husband, and wife. ... <u>The basic</u> <u>criterion</u>, then, for studying role behavior <u>is to identify the relevant social system or</u> <u>subsystem</u> and locate the recurring events which fit together in converting some input to some output. [Katz, Daniel and Kahn, Robert L., *The Social Psychology Of Organizations*, Wiley: New York, NY, 1966, p. 174]

Therefore, if Submitter, Tester, Programmer, Project Manager, Tester 1, Tester 2 or Test Lead. were to be regarded as roles, they must be <u>roles of individual users in a particular</u> <u>process</u> , rather than as <u>abstract roles associated with nodes representing abstract decision</u> <u>situations.</u> as recited in Claim 7.

Furthermore, the only place where Swenson even describes <u>more than one</u> <u>participant in an activity</u> is in connection with Figures 16 and 17, where it is clear that <u>each</u>

participant associated with an activity having multiple participants <u>has exactly the same role</u> (see

column 20, line 44 - column 21, line 4). Therefore, Swenson fails to describe modeling support for

participation of multiple persons in <u>differentiated</u> roles associated with a network node, as recited by

Claim 7.

Claims 8-9 are patentable at least for the reasons discussed above in that these

claims depend, directly or indirectly, from Claim 7.

The Rejection of amended, and now independent, Claim 15 is respectfully

traversed on grounds that it recites, "... using an object-oriented application framework to build

and configure decision process models comprised of interdependent decisions, ..." The Examiner

points to figure 2, which provides no description of the recited framework. As set forth in the

application [column 17, line 30 - column 18, line 12] ,

> "Abstract classes form the basis of a framework. If abstract classes factor out
> enough common behavior, other components, that is, concrete classes or other
> abstract classes, can be implemented based on the contracts offered by the abstract
> classes. A set of such abstract and concrete classes is called a framework.

> "The term *application framework* is used if this set of abstract and concrete classes
> comprises a generic software system for an application domain. Applications based
> on such an application framework are built by customizing its abstract and concrete
> classes. In general, a given framework anticipates much of a software systems's
> design. The design is reused by all software systems built with the framework.
> (Wolfgang Pree, *Design Patterns for Object-Oriented Software Development*,
> Addison-Wesley: Reading, MA, 1995, p. 54.)"

Swenson does not describe any such framework.

Claim 15 further recites, "... rendering said process models as directed graphs,

<u>whose nodes are concrete classes modeling decisions,</u> ..." The Examiner points to figure 2, which

describes a network, but the nodes of this and every other model network described in Swenson are

<u>object instances</u> rather than the recited <u>object classes</u>. Swenson states that, "The basic building

block of this tree [depicted in Fig.5] is <u>a C++ object that represent a stage of a work process plan</u>

<u>(e.g., an oval shown in FIG. 2)</u> [Column 11, lines 46-48]." The Examiner seems not to distinguish

between object <u>instances</u> and object <u>classes</u>, which, while related, are fundamentally different.

> "The concepts of a class and an object are tightly interwoven, …. However, there are important differences between these two terms. [Booch, Grady, *Object-Oriented Analysis and Design with Applications*, 2nd ed., Benjamin/Cummings: Redwood City, CA, 1994,p. 103]"

Therefore, while Swenson describes model networks with nodes that are object <u>instances</u>, it does not describe the recited network with nodes that are concrete <u>classes</u>.

Claim 15 further recites, "… rendering said project models as a <u>partition</u> of the graph of the instantiating process, where such partition is <u>defined by a specified node from the process graph and all and only those other nodes that are dependent on said specified node,</u> …" The Examiner points to columns 5-7. However there is no description there or anywhere else in Swenson which makes the recited distinction between a process model and a project model, nor any description of a graph that is a partition of another graph, much less the recited partition of claim 15.

The Rejection of independent Claim 17 is respectfully traversed on grounds that it recites, "… using a network or graph whose nodes are abstract decision situations representing choices to be made, which choices are <u>modeled by concrete decision classes</u> and by instances of those classes … providing arc objects <u>directed in each instance by an ordered pair of concrete decision classes</u> associated with each arc object, <u>where an entry or initial member of said ordered pair produces a data result required by an exit or terminal member of said ordered pair</u> [underscore added.]." To support his rejection, the Examiner points to columns 5-7, but nowhere in those columns does Swenson describe any such network. As argued above with respect to claim 15, the Examiner fails to distinguish between object <u>instances</u> and object <u>classes</u>. Swenson may describe a network whose nodes

are object <u>instances</u>, but nowhere describes a network whose nodes are concrete <u>classes</u>, much less the specific concrete classes recited by the claim. All of the networks described by Swenson (i.e., Figs 2, 2a, 5, and 6) have nodes that are <u>object instances only</u>, <u>rather than object classes</u> and instances of those classes as recited by claim 17.

Furthermore, Swenson does not describe a requirement for ordering the nodes of a network as recited by Claim 17. Swenson does describe a combination of "small circles [referred to as expectations]" and "arrows ... referred to as obligations," [column 5, line 62 - column 6, line 22 and Fig. 2, items 52-72 and 52a-72a]. Swenson asserts that, "An arrow leading from a small circle pointing to another stages [sic.] generally mean [sic.] that the stage being pointed to are [sic.] ready to be and must be completed after the expected course of action represented by the small circle is performed [column 6, lines 4-7]." Even if this were construed as a description of a directed arc, there is no description of <u>any basis for the direction</u>, much less the specific basis recited by Claim 17. Even where Swenson describes the creation of the described circles and arrows [column 22, lines 32-67 and Fig. 19, items 416-426] it is entirely silent with regard to a basis for any ordering denoted by the entities so created. Therefore, while Swenson may describe an ordering of the nodes of a network, it describes no basis whatsoever for such ordering, much less the specific basis recited by Claim 17.

Claims 18-20 are patentable at least for the reasons discussed above in that these claims depend, directly or indirectly, from Claim 17.

The Rejection of independent Claim 21 is respectfully traversed on grounds that it recites a requirement for, " An object-oriented application framework ... comprising (a) an abstract, extensible decision class ..., and an abstract, extensible data class ... , or alternatively, (b) a single abstract, extensible class which combines the attributes and

methods of said abstract decision and data classes," whereas, as argued above with respect to claim 15, Swenson describes neither a framework nor any extensible classes, much less the specific extensible classes recited by Claim 21. In support of his rejection, the Examiner points to Figures 1-2 and columns 5 -7. However, in the cited figures and text, Swenson does not even describe (1) an object class, much less (2) an abstract class and even less (3) an abstract, extensible class, and still less (4) any of the specific abstract, extensible classes recited by Claim 21. Swenson may elsewhere describe object classes, and perhaps even an abstract object class, but nowhere describes any extensibility with respect to a class. Nor does Swenson anywhere describe an application framework of any kind, much less an application framework with the specific components and arrangements recited by Claim 21.

Claims 22-29 are patentable at least for the reasons discussed above in that these claims depend, directly or indirectly, from Claim 21.

The Rejection of independent Claim 30 is respectfully traversed on grounds that it recites, "… constructing a computer-based process model for each of said one or more work processes, wherein <u>each said process model includes at least two instances of a first network</u>; [and] requiring that <u>each of said at least two instances of said first network be comprised of three or more nodes</u>; …" The Examiner points to Swenson's figure 5 to support his rejection of these claim elements. Swenson says that, "FIG. 5 is <u>a tree diagram of the organization of objects</u>, representing stages in a work process, stored in the system shown in FIGS. 3 and 4. [Column 3, lines58-60, underscore added]." Even if Figure 5 were to describe a process model composed of three or more nodes, it is nevertheless unclear how it describes including <u>two</u> instances of a network. Swenson's figures 2a and 6, might be seen to describe one, specific process model that happens to include at least two instances of a network comprised of three or more nodes. However, the

claim does not recite a mere possibility of two instances of a network in some process

model, but rather a requirement for each process model. Swenson fails to describe such a

requirement. Indeed, Swenson's Figures 2 and 5 actually teach away from the recited

requirement, by describing a process model comprised of a single network instance.

Claim 30 further recites, "… requiring that a first node of said three or more

nodes model an activity of one of said one or more work processes; requiring that a second

node of said three or more nodes model behaviors of a first role of a first participant in said

activity; requiring that a third node of said three or more nodes model behaviors of a second

role of a second participant in said activity;" Figure 5, cited by the Examiner, not only (1)

fails to describe the recited two instances of a network, but (2) also fails to describe any

node that models role behaviors of a participant in an activity modeled by another node of

that same network, (3) much less two such nodes modeling two such role behaviors as

recited by the claim. As argued above with respect to claim 7, the Examiner's rejection rests

upon failure to distinguish between a real world entity and an abstract entity. Claim 30 does

not recite that someone "has a role," but rather that particular nodes (i.e., a second node

and a third node) of a network, model behaviors of two distinct roles. Such modeling is

nowhere described in Swenson.

And, as also argued with respect to claim 7, what the Examiner alleges to be

described in Swenson as roles, are actually described in Swenson as individuals, and if they

were to be regarded as roles, they must be roles of individual users in a particular process ,

rather than two roles in an activity, much less roles in each of at least two activities in any

modeled process, as recited in Claim 30.

Also as argued above with respect to claim 7, the only place where Swenson

even describes more than one participant in an activity is in connection with Figures 16 and

17, where it is clear that <u>each participant</u> associated with an activity having multiple

participants <u>has exactly the same role</u> (see column 20, line 44 - column 21, line 4).

Therefore, Swenson fails to describe modeling of behaviors of <u>two or more roles in an</u>

<u>activity</u> as recited by Claim 30.

Claims 31-33 are patentable at least for the reasons discussed above in that

these claims depend, directly or indirectly, from Claim 30.

The Rejection of independent Claim 34 is respectfully traversed on

grounds that it recites modeling, "… participation of one or more persons <u>in said each of</u>

<u>said decision situations</u>, said participation being modeled <u>as at least two decision roles;</u>"

The Examiner points to Figures 2 - 6. However, there is nothing in the cited figures, nor

anywhere else in Swenson, which describes (1) a role of any sort, much less (2) the more

specific recitation of <u>decision</u> role, or (3) the yet more restrictive, <u>at least two</u> decision roles.

Nor is there anything in the cited figures, or anywhere else in Swenson, which describes (4)

the <u>modeling of roles</u>, much less (6) the modeling of decision roles <u>in said each of said</u>

<u>decision situations</u>. As noted above, in rebuttal of the rejection of Claim 30, the Examiner

asserts, "… that in Swenson, the Submitter or Tester or Programmer <u>has a role</u> because

each one is responsible for a task. [Underscore added]" However, the Examiner's assertion

again has no bearing on the issue of anticipation here, because Claim 34 does not recite that

someone "has a role," but rather that <u>participation</u> of persons in each decision situation <u>is</u>

<u>modeled as at least two decision roles</u>. Such modeling is nowhere described in Swenson.

Claim 34 further recites a requirement. " … that <u>each of said</u> at least two

decision <u>roles be associated with</u> said <u>each</u> of said <u>decision situations</u>; …" The Examiner

again points to Figures 2 - 6, which contain nothing that describes this recited claim

element.

Claim 34 further recites a requirement. " ... that said each of said at least

two decision roles <u>have defined behaviors</u>; ..." The Examiner cites Swenson's Figure 5,

which Swenson says, " ... is a tree diagram of the organization of <u>objects, representing</u>

<u>stages in a work process</u>, stored in the system shown in FIGS. 3 and 4. [Column 3, lines

58-60]." Swenson describes neither (1) a decision role, nor (2) any behaviors of any such

role, much less (3) the more specific arrangement of those entities which is recited by Claim

34.

Claim 34 recites a further requirement, " ... that said defined <u>behaviors</u> of

said each of said at least two decision roles <u>be differentiated from</u> said defined behaviors of

<u>every other one</u> of said at least two decision roles; ..." The Examiner points to Column 13,

line 23 - column 14, line 40, where Swenson says that, "Each of the sub-colloquies are

themselves [software] agents, containing [1] a short and long description, [2] name, [3] <u>a</u>

<u>responsible user or group of users</u> and [4] an action table of expectations and

corresponding obligations [Column 13, lines 60-63, underscore added]." The Examiner

may be arguing with this citation that Swenson here describes a "responsible user" <u>role</u>.

However, Swenson's use of the term *responsible user* is far more modest.

> "When the symbol for a stage is positioned, in block 412 (CCM), the
> colloquy context creates a VplActiveAgent C++ object (agent), such as that
> described in reference to Table 1, representing the stage.... , <u>the system</u>
> <u>prompts for and accepts from the user</u>, a short description of the stage, such
> as "Can the Problem be Fixed? 46a, a long description of the stage that
> appears on the viewer context screen (e.g., "Can you fix the problem, or
> should this report be deferred or rejected?" in FIG. 9), <u>and [the identity of]</u>
> <u>the user or users responsible for the stage, such as Programmer 46b</u>. The
> short description is stored in ShortDescription 460, the long description is
> stored in LongDescription 462 and <u>[the identity of] the user or users</u>
> <u>responsible for the stage is stored in *Groups 464</u> [column 22, lines 5-23,
> underscore added; numerical references are to items in Fig. 19 or Table 1 in
> column 25]."

It should be clear from the above description that, rather than describing a

role as recited by the claim, Swenson merely uses the term *responsible user* <u>to refer a</u> <u>instance variable</u> of a VplActiveAgent C++ object (agent), which instance variable <u>simply</u> <u>points</u> to one or more <u>individual users of the object</u> containing that instance variable.

       Even if one were to grant that Swenson describes a responsible user role, it still fails to describe the recited claim because it would describe <u>at most one</u> role while the claim recites "<u>at least two</u> roles," and goes on to recite additional requirements nowhere described in Swenson, including (1) that each decision role be <u>associated with each decision</u> <u>situation</u>, (2) that each of the at least two roles have <u>defined behaviors</u>, (3) that the defined behaviors of each decision role be <u>differentiated</u> from those of every other decision role, and (4) that the defined behaviors of each decision role be <u>invariant</u> with respect all decision situations.

       Claims 35-45 are patentable at least for the reasons discussed above in that these claims depend, directly or indirectly, from Claim 34..

       The Rejection of independent Claim 46 is respectfully traversed on grounds that it recites, "… constructing a computer-based process model of each of said one or more work processes, wherein <u>each said process model includes a network with a</u> <u>concrete object class at each node</u> of said network; …" The Examiner points to four extended sections of text and Figures 2 -6 in support of his rejection. What he sees there that describes the recited claim element is completely unclear. Swensen may describe <u>either</u> a process or a project model, "each of which includes a network," and does describe a concrete object class, but Swenson does not describe the more specific recitation of the claim, because none of the networks described by Swenson contains (1) an object <u>class at</u> <u>any node</u>, much less (2) an object <u>class at each node</u>.

       Claim 46 further recites: " … a <u>customizable</u> object class … to model a

work element of any one of said one or more work processes; generating said concrete

object class at each said node of each said process model by customizing said customizable

object class; generating ... project models from each said... process model, wherein each of

said one or more project models includes a network with an object instance of a concrete

object class at each node; [and] requiring that each said object instance at the node of any of

said one or more project models be an instance of said concrete object class at a

corresponding node of said process model from which said project model has been

generated; ..." It is again unclear why the Examiner points to the same four extended

sections of text and Figures 2 -6. Swenson may describe either process models or project

models that are networks with an object instance at each node of the network. However, in

addition to there being no description in Swenson of (1) a modeled network with a concrete

object class at each node, there is nothing in Swenson that describes (2) a customizable

object class, nor (3) the generation of the object instances at the nodes of a network by

customizing a class, nor (4) of one type of model being generated from another type of

model, nor (5) of the recited requirement for correspondence between elements of such

models. Swenson thus fails to describe at least five elements of the recited claim.

Claims 47-55 are patentable at least for the reasons discussed above in that

these claims depend, directly or indirectly, from Claim 46.

Claims 40 and 53 stand rejected under 35 U.S.C. §103(a) as

unpatentable over Swenson. These rejections are respectfully traversed on a basis

that the Examiner's rationale for rejection reflects an erroneous reading of Claims 40 and

53. The Examiner asserts that, "... It would have been obvious ... to know that it would be

good business practice to mark documents with trademarks or copyright logos to indicate

protection."

The Examiner reads these claims as requiring "that a copyright notice and trademark be displayed on said document, ..." However, Claims 40 and 53 recite no such requirement. Rather, they narrow the requirement of Claims 39 and 52 respectively by specifying more precisely the "predetermined criteria" recited in Claims 39 and 52, that are to be used in a <u>context</u>, "... requiring that a copyright notice and trademark [notice] be displayed ..."

Reference to trademark and copyright notices as used in claims 40 and 53, <u>does not</u> establish a requirement for a copyright or trademark notice. Rather, it specifies more fully the required behavior of the inspector role in <u>a specific context that happens to be characterized</u> by a requirement for trademark and copyright notices. That the display of copyright and trademark notices is common is not disputed. Indeed it is that commonality that makes it useful to cite it as an noteworthy condition. But there is no recitation of any requirement for the display of either a copyright notice or a trademark as the Examiner asserts. Therefore, whether the use of trademark and copyright notices is or is not "good business practice" is not relevant to the claim.

Conclusion

For all the reasons advanced above, Applicant respectfully submits that the application is in condition for allowance and that action is earnestly solicited.

AUTHORIZATION

The Commissioner is hereby authorized to charge any additional fees which may be required for this amendment, or credit any overpayment, to a credit card as authorized by the executed Credit Card Payment Form (PTO-2038) submitted herewith.

In the event that an extension of time is required, or may be required in addition to that requested in a petition for an extension of time, the Commissioner is requested to grant a petition for that extension of time which is required to make this response timely and is hereby authorized to charge any fee for such an extension of time or credit any overpayment for an extension of time to a credit card as authorized by the executed Credit Card Payment Form (PTO-2038) submitted herewith.

Respectfully submitted,

Paul M. Konnersman
Applicant, *Pro Se*

272 Ocean Avenue
Marblehead, MA 01945-3730
781-639-0616
konnersman@comcast.net